Temporal Abstraction in Bayesian Networks

Brendan Burns and **Clayton T. Morrison**

University of Massachusetts
Amherst, MA 01003
{bburns,clayton}@cs.umass.edu

Abstract

A current popular approach to representing time in Bayesian belief networks is through Dynamic Bayesian Networks (DBNs) (Dean & Kanazawa 1989). DBNs connect sequences of entire Bayes networks, each representing a situation at a snapshot in time. We present an alternative method for incorporating time into Bayesian belief networks that utilizes abstractions of temporal representation. This method maintains the principled Bayesian approach to reasoning under uncertainly, providing explicit representation of sequence and potentially complex temporal relationships, while also decreasing overall network complexity compared to DBNs.

Introduction

Time is a critical element for reasoning in many problem domains. A war-gaming system must analyze and predict an enemy's tactics and intentions as they unfold in behavior over the duration of a battle. An autonomous agent exploring its world needs to discover temporal relationships between its actions and changes in its environment. In fact, the core of any planning system involves reasoning about temporal sequence. Often such reasoning takes place at numerous levels of abstraction above temporal snapshots. For example, a robot tasked with understanding the affects of its actions in the service of obstacle avoidance should not reason over only the sequence of instantaneous temporal representations: "at time t my motors started forward," "at time $t + k$ I hit a wall." Instead, it needs to represent the fact that "hitting a wall was preceded by moving forward." This latter representation not only makes the relationship between events explicit, it seems to be closer to the way humans represent their world as well.

At the same time, real world domains inherently involve uncertainty. Sensors are noisy and we don't have access to all of the states of the world (including the mental states of others, or even the outcomes of our own actions). Bayesian belief networks (Pearl 2000) provide a principled framework for reasoning about uncertainty. But to handle uncertainty, we not only need to reason about uncertain states, we must also consider how those states change over time. A current popular approach to incorporating temporal representation into Bayesian belief networks will be described below; unfortunately the representation necessitates only reasoning between instantaneous moments.

A great deal of progress has been made in formally representing temporal relationships. One approach is *fluents*, which is based on Allen's temporal relations (Allen 1981). Fluents represent the pairwise temporal relationships between propositions with temporal extent. Figure 1 shows the six basic fluent relationships. Armed with the ability to represent these relationships, a robot could represent the event of moving forward and then hitting a wall using, e.g., the fluent relationship ES, representing *moving forward* as ending at the same time *contact* with a wall begins.

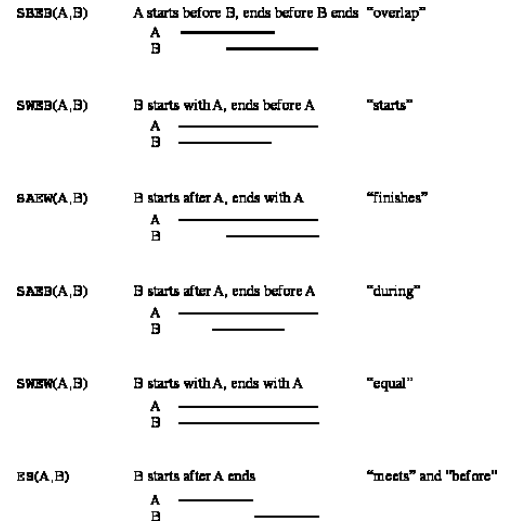|		
SBEB(A,B)	A starts before B, ends before B ends	"overlap"
SWEB(A,B)	B starts with A, ends before A	"starts"
SAEW(A,B)	B starts after A, ends with A	"finishes"
SAEB(A,B)	B starts after A, ends before A	"during"
SWEW(A,B)	B starts with A, ends with A	"equal"
ES(A,B)	B starts after A ends	"meets" and "before"

Figure 1: Six base fluent relationships.

In this paper we propose a template for structured temporal reasoning in Bayesian networks. Two techniques are proposed which incorporate increased abstraction of temporal sequence and making temporal relationships explicit through the use of fluent relations.

The rest of this paper is organized as follows. The remainder of this section briefly describes Dynamic Bayesian Net-

Report Documentation Page

Public reporting burden for the collection of information is estimated to average 1 hour per response, including the time for reviewing instructions, searching existing data sources, gathering and maintaining the data needed, and completing and reviewing the collection of information. Send comments regarding this burden estimate or any other aspect of this collection of information, including suggestions for reducing this burden, to Washington Headquarters Services, Directorate for Information Operations and Reports, 1215 Jefferson Davis Highway, Suite 1204, Arlington VA 22202-4302. Respondents should be aware that notwithstanding any other provision of law, no person shall be subject to a penalty for failing to comply with a collection of information if it does not display a currently valid OMB control number.

1. REPORT DATE **2003**	2. REPORT TYPE	3. DATES COVERED **00-00-2003 to 00-00-2003**

4. TITLE AND SUBTITLE	5a. CONTRACT NUMBER
Temporal Abstraction in Bayesian Networks	5b. GRANT NUMBER
	5c. PROGRAM ELEMENT NUMBER
6. AUTHOR(S)	5d. PROJECT NUMBER
	5e. TASK NUMBER
	5f. WORK UNIT NUMBER

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES) **University of Massachusetts,Department of Computer Science,Amherst,MA,01002**	8. PERFORMING ORGANIZATION REPORT NUMBER

9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)	10. SPONSOR/MONITOR'S ACRONYM(S)
	11. SPONSOR/MONITOR'S REPORT NUMBER(S)

12. DISTRIBUTION/AVAILABILITY STATEMENT
Approved for public release; distribution unlimited

13. SUPPLEMENTARY NOTES

14. ABSTRACT

15. SUBJECT TERMS

16. SECURITY CLASSIFICATION OF:			17. LIMITATION OF ABSTRACT	18. NUMBER OF PAGES **6**	19a. NAME OF RESPONSIBLE PERSON
a. REPORT **unclassified**	b. ABSTRACT **unclassified**	c. THIS PAGE **unclassified**			

works (DBNs) (Dean & Kanazawa 1989), the current popular approach to handling time in Bayes belief networks. Sections and present two temporal abstractions for Bayesian networks. The first allows for efficient and descriptive combination of temporal data at multiple levels of granularity. The second represents the more abstract fluent relations. In Section , illustrative examples of the usage of these two temporal abstractions are presented. Finally, related work and future directions are discussed in Sections and , including our use of this method in our Bayesian blackboard system, AIID (Architecture for the Interpretation of Intelligence Data).

Time in Dynamic Bayes Networks

A DBN represents a system as a sequence of snapshots, from times $t_1 - t_N$. Each snapshot consists of a complete network structure representing the state of the system at that time (see Figure 2). Causal links are added between the nodes of sequential timesteps, representing the sequential relationship from t_k to t_{k+1}. Dynamic Bayes networks were first utilized by Dean and Kanazawa (Dean & Kanazawa 1989) and Nicholson (Nicholson & J.M. 1992). A significant discussion of DBNs and robotic motion planning is given in (Dean & Wellman 1991).
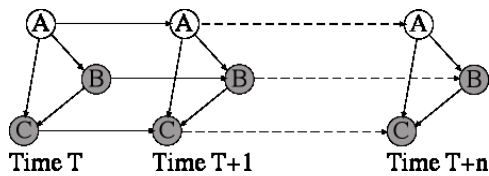

Figure 2: A generic Dynamic Bayes Network

In practice, most DBN implementations assume for the sake of efficiency that the Markov property holds for the domain they represent. This means that representing a single snapshot in the past is sufficient for predicting future outcomes. By making this assumption, all of the nodes in the past may be "rolled up" into a single network of nodes which are connected forward to the network representing the "present." This can lead to a significant reduction in over-all network complexity and subsequent performance enhancement.

In problem domains where the granularity of temporal representation is constant throughout, the above approach to representation makes sense. However, when granularity of time varies throughout and we wish to represent variables whose states depend on different time scales, some network nodes may be repeated redundantly in numerous instantaneous networks. Consider an example of a signal processing task in which a signal is sensed at regular intervals, but whose interpretation is based on a significantly longer duration. In traditional DBNs, the node representing this interpretation is replicated in each instantiation of the network representing one time slice. This redundant replication of the node slows down computation, complicates reasoning in the network and makes human interpretation of the networks difficult.

DBNs also restrict knowledge engineering. There is no way in a DBN to represent the concept "A starts with B but ends before B ends." In fact there is really no way to represent as simple a concept as "A comes before B" in such a way that another node can be causally related to that statement.

These limitations of DBNs pose two problems to be solved. Networks need to be able to represent time at different levels of granularity and abstract temporal relationships should be easily expressible in the Bayesian belief network framework. Fortunately, providing for different levels of temporal granularity in Bayes nets facilitates the incorporation of abstract temporal relationships.

The following section describes how hierarchical Bayes networks can be used to provide different levels of temporal granularity. Section shows how abstract temporal relationships, in this case fluents, can be incorporated into a Bayes network. Examples of such networks are presented in Section .

Reasoning over multiple levels of temporal granularity

Imagine an autonomous agent exploring an environment while attempting to avoid obstacles. It receives sonar inputs every fifth of a second and front bump sensor indications once each second. The agent would like to perform reasoning about its collisions with obstacles. Obviously the agent can not perform this reasoning more frequently than once a second. Thus the agent has two levels of temporal granularity it needs to reason over. Traditional DBN techniques would resolve this difference in granularity by replicating the same bump sensor value across five different nodes in the dynamic Bayes network (figure 4). The solution we propose incorporates all five sensor values (and their related networks) into the causes of a single node at the larger granularity.

Hierarchical Bayes Networks

Hierarchical Bayes networks (Koller & Pfeffer 1997) have been proposed as a method of incorporating a Bayes network "inside" a node in another Bayes network. In practice this means that the only point of causal contact between two sub-networks is a small subset of the nodes shared by both sub-networks. The advantage of this technique is both in computational speed, since the number of possible combinations of outcomes to examine is reduced, and knowledge engineering, since such encapsulation facilitates the hierarchical decomposition of concepts. We adopt this approach by using a single node at a coarse temporal granularity to encapsulate a number of finer grained temporal nodes.

Building temporal abstractions

To incorporate fine grained information at a coarse time scale we add a single node at that coarse scale. This node is linked into the network as the cause of the finer grained nodes. Each link represents the probability of the finer grained node given the coarse one ($P(n_{t_{fine}}|n_{t_{coarse}})$.
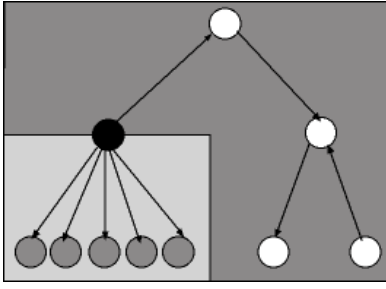
Figure 3: An example of the temporal hierarchical network.

It can be seen in Figure 3 that if the coarse node is used only as a cause on the coarse scale, it d-seperates the finer grained information from anything on the coarser scale. In the figure, the temporal nodes in the light grey box are d-separated from the rest of the network (dark gray box) by the black temporal abstraction node. This separation means that inferences about network state "above" that coarse-grained node need only consider the state of that node, not the finer-grained nodes associated with it. This restriction is not necessary from a theoretical stand-point, but it allows for inference algorithms to run more efficiently on the network. This process of temporal abstraction is not limited to just two levels of temporal granularity; the same process can be repeated to abstract an arbitrary number of different temporal granularities to a single level.

Conditional probability tables in hierarchical networks
Once the network structure illustrated in Figure 3 has been constructed, it is necessary to fill in the conditional probability tables for each of the nodes at the finer temporal grain, n_{t_f}. We need a function f which calculates the probability of each fine grained node n_{t_f} given the value of the coarse grained node n_{t_c}, e.g. $P(n_{t_f}|n_{t_c})$. For our purposes we will define f as follows:

$$\begin{aligned} f(n_{t_f} = j, n_{t_c} = k) &= 1 - \alpha, j = k \\ &= \alpha, j \neq k \end{aligned}$$

This function models a fine grained temporal sensor which makes an erroneous observation with probability α. As an example of this model, imagine that we know a robot is pressing against an object for some temporal extent t'; the probability of some instantaneous sensor observation within t' returning false is α, the probability of the sensor malfunctioning. Likewise, if the robot is *not* pressing against an object for some temporal extent t'', then the probability of an instantaneous observation at some time within t'' returning true is the same α (assuming that malfunctions are uniformly distributed between false positives and false negatives). Note that this conditional probability function assumes that n_{t_f} has discrete values. The case of continuous valued nodes is significantly more complex since the semantics of the temporally abstracting node are less clear. It might be the average continuous value over its temporal extent, or any other arbitrary function. As a result of these complexities (and the fact that our temporal calculus, fluents, is con-

structed over boolean values) we assume our temporal nodes have discrete values.

Adding Fluents to Bayes Networks

As described earlier, fluents are a set of pairwise relationships between predicates whose values have temporal extent. Or, equivalently, they are a set of pairwise relationships between the starts and ends of changes in truth value of boolean variables. To incorporate fluents into Bayesian networks we re-represent the sequence of instantaneous representations of boolean variables by turning them into temporal distributions over the time interval during which the variable was true. After this transformation, we link these start and end nodes as (potential) causes of a particular fluent. The following two sections describe this process in more detail.

Adding Start and End Nodes

Start and end nodes are added in much the same way as the temporal-granularity-abstracting nodes described in Section . However, instead of a node abstracting to a coarser level of temporal granularity, a single node, with a distribution over the time for which a boolean variable becomes or stops being true is added to the network. This node is linked into the network as a cause of the instantaneous true or false values across time slices. Given this topology, we can specify the conditional probability table of each instantaneous true or false node as follows. Given a particular start time t, the probability of the node at time t being true is 1.0, and the probability of node $t + n$ is $1.0 - K^n$ where K is the probability that a node will switch from true to false. For performance reasons when building the network, start and end nodes should only be connected to instantaneous nodes at points in time when it seems likely that a start or end might occur. In hand crafted networks this would be the task of a knowledge engineer, alternatively, automated methods, such a leading or trailing edge detector could search for such boundaries and place start and end nodes.

Introducing Fluents

Now that we have a way to abstract instantaneous time to starting and ending nodes of an interval, these starts and ends can be utilized for higher level reasoning. Fluents, introduced in Section , give us a complete specification of the temporal relationships between beginnings and ends of event episodes (in this case, change in boolean variable truth values). Given two pairs of event beginnings and endings it is easy to give the probability of a particular fluent relationship F existing between the two events A, B. $P(F(A,B)|Start(A), End(A), Start(B), End(B))$ is 1.0 when the fluent relation holds for the two events and 0.0 otherwise.

Examples

In the following sections we present two examples of temporal abstractions, building from time-slice representation to representation of time intervals, and then to representation of relations between intervals. In these examples, we again

consider an autonomous mobile agent attempting to reason about events in its environment, including the outcomes of its own actions. This agent is equipped with two sensors, A and B. Sensor A reports true when the agent is getting closer to an obstacle. Sensor B is a bump sensor which reports true when it is in contact with an obstacle. Sensor A reports at 5hz and sensor B reports at 1hz. The first example illustrates reasoning about information with different temporal granularity. The second example builds on the first example, describing how fluents might then be incorporated to promote more sophisticated reasoning about temporal relations. In both cases the agent is reasoning about whether or not it is approaching and subsequently colliding with an obstacle (e.g., a wall).

Hierarchical temporal abstraction example

Using a traditional DBN, each time tick at the finest granularity (in this case 5hz) would instantiate a separate copy of the reasoning network, causally linked to the immediately previous network. Such a network is shown in Figure 4. Using the hierarchical representation described above, a network would instead be instantiated which looks like Figure 5.
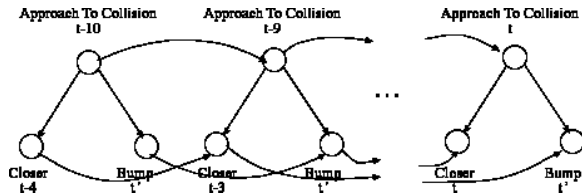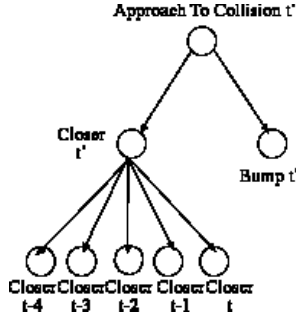

Figure 4: The dynamic Bayse network for example 1

Figure 5: The multiple temporal granularity network for example 1

Comparing the networks

As can be seen in Figures 4 & 5, the new formulation for abstracting time results in a more intuitive network structure: information contributing to distance is encapsulated and separated from information about the object's identity. Intuitive structure facilitates network engineering and analysis.

Our abstraction results in a faster network evaluation as well. To evaluate the speed of the networks we used an

Network Structure	Construct J-Tree	Query J-Tree
D.B.N.	388.9	23.0
Temporal Abstraction	194.2	15.5

Table 1: Junction tree run times for DBNs and Hierarchical Temporal Abstraction

implementation of the junction tree algorithm (Lauritzen & Spiegelhalter 1988). The DBN we used was one in which the Markovian assumption had been used to roll up the network so that the DBN contained only two instances of the network; current and past. As mentioned above, this the preferred usage of DBNs since it speeds querying significantly. For our hierarchical temporal abstraction we used the network in Figure 5. The results of this performance comparison are reported in Table 1. The results consist of averages of ten runs of construction and querying. It can be seen that our hierarchical temporal abstraction significantly outperforms the DBN in the construction of the junction tree and slightly outperforms the DBN in querying. It seems likely that the faster performance of the hierarchical temporal abstraction network is due to its simpler network structure and smaller conditional probability tables. In particular, the abstracting node for distance is able to d-separate the fine-grained samples from the rest of the network, which speeds inference.

The performance gains described above are even more significant because the DBN is required to perform the roll-up step at each tick of the finer temporal granularity, while our temporal abstraction only needs to query when the high-level probability is needed by the network's user not purely for proper maintenance of the priors in the network. In situations where the sampling rate of the sensors is quite fast, DBNs simply could not perform the roll-up step in real-time while the temporal abstractions presented would be able to keep up.

This first example illustrates both the increased clarity in knowledge engineering of temporal abstraction as well as the performance benefits in querying that the resulting network structure provides. We now turn to incorporating representation of temporal relations between sensor streams.

Fluents Example

The following example illustrates the use of fluents as a temporal abstraction in Bayes nets. Again consider an agent determining if it has moved forward resulting in a collision with an obstacle. In this case, however, the agent will use fluents to represent the temporal relationship between its distance and bump sensor. A fluent representing this behavior is given in Figure 6. The fluent is represented as SAEW($Closer, Bump$) (where SAEW is short-hand for Starts After Ends With; see Figure 1).

We can begin to build our Bayes net by representing the probability that the agent running into an obstacle will lead to the fluent represented above ($P($SAEW$(Closer, Bump)|ApproachCollision))$. This describes the network fragment given in Figure 7.

Given this network, we estimate the probabil-
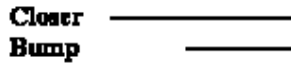
Closer ——————————

Bump   ——————————

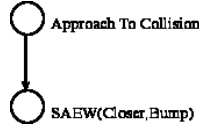Figure 6: SAEW fluent relationship between Closer and Bump

Figure 7: Initial Bayes network

ity of the fluent given particular starting and ending times of the two predicates; Closer and Bump (e.g. $P(CloserStarts|\text{SAEW}(Closer, Bump))$). This leads to the network in Figure 8.
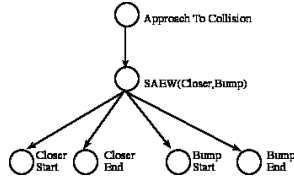

Figure 8: Interim Bayes network

To complete the network we need to ground it in instantaneous predicates. Using the same hierarchical temporal abstraction method in the first example, a starting or ending distribution for a particular predicate implies a probability for a particular observation at a moment in time, e.g. $P(Closer_t|CloserStarts, CloserEnds)$. Thus the final network constructed is seen in Figure 9.

Although the network depicted in Figure 9 is slightly larger than that shown in the first example (fig. 5), it can be seen that in this network the concept of approaching-to-collision is not parameterized by a time step. The network is no longer simply estimating the probability of approach-to-collision at a particular moment, but rather the probability that an approach-to-collision has occurred at all in the agent's history. Once it has been determined that such an event has occurred, the probabilistic contributions of the various nodes can be analyzed to determine the precise time frame in which it occurred.

Related Work

There have been many different efforts to integrate temporal reasoning in a Bayesian network framework. The relationship to dynamic Bayes networks has been discussed earlier in great detail. In addition, Berzuni (Berzuni 1990) proposed adding a number of individual nodes representing each temporal interval of interest as a random variable. Like the placement of start and end nodes discussed above, the placement of these temporal intervals must be carefully regulated or else it can significantly increase the size of the net-
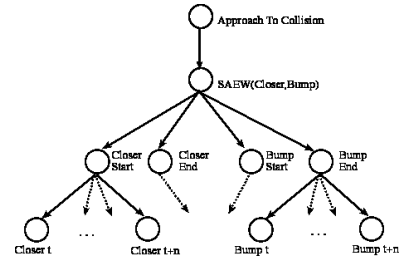

Figure 9: Complete Bayes network

work. Tawfik and Neufeld (Tawfik & Neufeld 1994) instead present a formulation where the conditional probability tables of nodes are defined as functions over time. Such a formulation requires exogenous knowledge of how the probabilities decay over time. For complex conditional probability tables such a function may be quite difficult to describe. Additionally it requires the network to keep track of each node's value over time and the time at which any particular observation is made. Santos and Young (Santos & Young 1999) propose extending the structure of belief networks to include time so that each node carries a value for a set of proscribed intervals and the arcs between nodes contain temporal extent. However they do not present a technique for inference in their extended networks which severely limits their usefulness in this context. All of these techniques concern themselves with defined intervals, either discrete or continous, none of them allow for the reasoning about sequence without specific temporal values.

Many people have done work on reasoning with fluents or other forms of temporal logic. Work has also been done on unsupervised algorithms to learn fluents from time series of boolean data (Cohen 2001; Cohen & Burns 2002).

Also related is work in object-oriented or hierarchical Bayesian networks (Koller & Pfeffer 1997) which form the basis for the construction of the time abstracting subnetworks for reasoning at different levels of temporal granularity.

Future Work

We are currently developing these approaches to representing time to incorporate temporal reasoning in a Bayesian blackboard system called *AIID: an Architecture for the Interpretation of Intelligence Data*. AIID composes smaller Bayesian network fragments, under the control of blackboard knowledge sources, to incrementally construct a complete belief network on the blackboard.

This paper presents our preliminary investigations, and we plan to expand these methods in a number of directions. First, fluent relationships are just one form of temporal abstraction and there are others which could be incorporated in a Bayesian network framework. Having multiple representations available is useful for our blackboard system (e.g., as independent knowledge sources), but more generally, it provides a knowledge engineer with a toolbox of temporal abstractions to use. Second, we are interested in extending network construction algorithms to autonomously identify

appropriate situations for hierarchical abstraction and fluent relation construction. Finally, as discussed above, the technique for performing abstraction over multiple temporal granularities is limited to discrete valued nodes. The estimation of the conditional probability of fine grained continuous nodes given a continuous coarse grained parent should be explored further since in many cases continuous representations may be more appropriate.

Conclusions

Two complementary techniques for incorporating time into Bayesian networks have been presented. Initial results indicate that the techniques result in simpler, more comprehensible networks. They constitute a progression of increasingly sophisticated representations of time, from time-slice snapshots, to intervals, to fluent relationships, while remaining within the principled framework of Bayesian belief networks.

Acknowledgments

References

Allen, J. F. 1981. An interval based representation of temporal knowledge. In *IJCAI-81*, 221–226.

Berzuni, C. 1990. Representing time in causal probablistic networks. In *Uncertainty in Artificial Intelligence Five*.

Cohen, R., S. C., and Burns, B. 2002. Learning effects of robot actions using temporal associations. In *Proceedings of the 2nd International Conference on Development and Learning*, 96–101.

Cohen, P. R. 2001. Fluent learning: elucidating the structure of episodes. In *Proceedings of the Fourth Symposium on Intelligent Data Analysis*, 268–277.

Dean, T., and Kanazawa, K. 1989. A model for reasoning about persistence and causation. *Computational Intelligence* 5(3):142–150.

Dean, T. L., and Wellman, M. P. 1991. *Planning and Control*.

Koller, D., and Pfeffer, A. 1997. Object-oriented bayesian networks. In *Uncertainty in Artificial Intelligence: Proceedings of the Thirteenth Conference (UAI-1997)*, 302–313. San Francisco, CA: Morgan Kaufmann Publishers.

Lauritzen, S. L., and Spiegelhalter, D. J. 1988. Local computations with probabilities on graphical structures and their application to expert systems. *Journal of the Royal Statistical Society* 50(157):224.

Nicholson, A., and J.M., B. 1992. The data association problem when monitoring robot vehicles using dynamic belief networks. In *ECAI 92: 10th European Conference on Artificial Intelligence*.

Pearl, J. 2000. *Causality: Models, Reasoning and Inference*. Cambridge University Press.

Santos, Jr., E., and Young, J. D. 1999. Probabilistic temporal networks: A unified framework for reasoning with time and uncertainty. *International Journal of Approximate Reasoning* 20(3):263–291.

Tawfik, A. Y., and Neufeld, E. 1994. Temporal bayesian networks. In *Proceedings of First International Workshop on Temporal Representation and Reasoning (TIME)*.